```
 AAA    TTTTT   AAA    RRRR    III        ****          SSS    TTTTT
A   A     T    A   A   R   R    I         ****        S     S     T
A   A     T    A   A   R   R    I         ****        S           T
AAAAA     T    AAAAA   RRRR     I        ******         SSS       T
A   A     T    A   A   R   R    I      **  **  **          S      T
A   A     T    A   A   R   R    I     **   **   **    S     S     T
A   A     T    A   A   R   R   III   **    **    **     SSS       T

DDDD   EEEEE  V   V  EEEEE L      OOO   PPPP   EEEEE RRRR    SSS
D   D  E       V V   E     L     O   O  P   P  E     R   R  S   S
D   D  E       V V   E     L     O   O  P   P  E     R   R  S
D   D  EEEE    V V   EEEE  L     O   O  PPPP   EEEE  RRRR    SSS
D   D  E        V V  E     L     O   O  P      E     R   R      S
D   D  E        V V  E     L     O   O  P      E     R   R  S   S
DDDD   EEEEE     V   EEEEE LLLL   OOO   P      EEEEE R   R   SSS
```

Question and Answer Bulletin   February 1986
Copyright (C) 1986 by Atari Corp.   "all rights reserved"
1196 Borregas Ave.,   Sunnyvale, Ca.   94086

Here are the latest questions from the Atari developers
mailbag as answered by John Feagans, Director of Software
Technology. Leave questions on Compuserve for PIN 70007,1072
or GO PCS57 for Atari developer SIG information.

Q: How can I have a different background color for text.
There does not seem to be anything I can set to make this
happen.

A: A way we recommend to make this happen is to do a
rectangle fill of the area behind the text then use v_gtext
with a transparent writing mode.

Q: I have a GEM application that I am porting from the IBM
PC. Everything compiles O.K. but I get a number of undefined
symbols in the link step. Some of these symbols are
dos_free, dos_gdrive, dos_lseek, and dos_open.

A: The problem here is some differences in name between what
you are using, what is described in the GEMDOS spec, and what
is actually in the bindings. Here is a table which may
explain what is happening:

Your code	GEM DOS manual	osbind.h
dos_free	d_free	Dfree
dos_gdrive	d_getdrv	Dgetdrv
dos_lseek	f_seek	Fseek
dos_open	d_open	Fopen

Q: From the VDI there doesn't seem to be a call to set pixels. How do I do it and be compatible with all resolutions?

A: You can do it by drawing a 1 pixel wide line with a length of 1 pixel. There is also a line-A call for setting pixels. The latter method is much faster, however, it does not clip.

Q: I just installed my ROMs and the system doesn't come up. It is just a white screen. It is like this every time I reset. Oh...there it is...what happened?

A: If you have just installed your ROMs and do not have a disk drive attached, the TOS is going to spend considerable time attempting to read some files from disk. It is looking for desk accessories, an auto folder which may contain drivers to load, and a desktop.inf file created if you saved a desktop. If you want to shorten the power-up process, then have a disk attached with a valid floppy disk in the drive.

Q: I just ported my IBM PC version and have the symbol UMULDIV undefined.

A: VDIBIND defines MUL_DIV, mul_div, and umul_div, but the symbol UMUL_DIV is missing. If you are programming in C perhaps you could #define it in a header file. In assembler, you could equate UMUL_DIV to umul_div.

Q: Can I boot another operating system if I have TOS in ROM?

A: Yes. The system has been designed so that a disk with a boot sector can have another operating system which can be loaded in. Another way to load another system would be to put a driver in the auto folder on the system disk. A third interesting method would be to make a desk accessory which loads the new system. It is also possible for a program to perform an extended bios call PUNTAES which frees up the memory used by the AES and desktop while preserving the bios,dos, and VDI.

Q: How can I load in code from BASIC? Where is it put? How do I allocate space for it?

A: To use the BLOAD function you must first allocate a string array big enough to hold your code. Next use the varptr function to get the address of the storage area. Use BLOAD with this address and the CALL function to execute your code. One final point--make sure the code is relocatable and completely relative as BLOAD does no fix-ups.

Q: How are the icon outlines dragged on the desktop and what is the shape?

A: Icons are dragged on the desktop by moving a polyline XOR box. The vertices are a fixed set that resemble the edges of the mask data for the icon.

Q: How do I stop BASIC from drawing windows on my output screen?

A: Windows are redrawn when BASIC calls a routine in the event library. Some examples of these routines are evnt_multi, evnt_keyboard, and evnt_button. It is possible to stop the evnt_multi calls because there is a BASIC system table called SYSTAB. Check the documentation for particulars. At an offset of +24 there is a variable called GEMFLAG. A POKE of 1 in this location turns the event calls off and a POKE 0 turns the event calls on.

Q: When I double click on AS68 from the desktop, the screen turns white and immediately returns to the desktop. How do I type in the name of the file to assemble?

A: It is posible to run AS68 from the desktop. The procedure is to install it as a TTP type, (TOS takes parameters) and save desktop, or show-info and rename it as a .TTP. Next time you double click on AS68, a dialog box for parameters will come up. You can then type "-u -l myfile.s" and click on OK to run the program. Object files fill be generated and control returns to the desktop upon completion of the assembly.

Q: I have a program where I am drawing my own dialog boxes and menu by constructing the object tree and calling OBJC_DRAW. The problem is that when I draw the dialog box, the menu pops down. I have tried everything to prevent this. What is happening?

A: We suspect what is happening here is that you may be passing the address of the menu rather than the dialog box. If you have the level set to MAXDEPTH, everything including the dialog box over a popped down menu may be drawn. Double check what you are passing to the objc_draw.

Q: I have a dialog box with an editable field. When I try to put string to initialize the data, everything is messed up. It seems like the pe_text pointer is pointing to the wrong place!

A: The OBSPEC field of the editable object points to a TEDINFO structure. In this structure there are three pointers: in order they are 1) to a string containing the actual text, 2) to the template, and 3) to a character validation field. Beyond getting the proper pointer, another problem you may encounter is that the string is a fixed length. You may be moving data into it which is longer than was created at the time you defined the text in the resource

construction set.

Q: I am running the batch program and get a bus error after
c0.prg has been running a little while compiling my program.
Could this be a bug in the compiler.

A: A bug is always possible but here are a few other things
to suspect when a problem like this occurs. The batch
program will abend if it cannot find a specified program to
load.   You may also have an include file with something
illegal.   That might explain why things proceed through the
pre-processor in abort in the first compile step. Another
thing to check is if you are running out of space on the disk
from which the compiler is running. Each step of the compile
creates some intermediate files which require space. It is
possible to specify to the compiler which drive to which it
writes the intermediate files.

Q: I am transferring files from my IBM PC to the ST using
kermit.   ASCII files come over o.k. but binary files are
messed up.

A: When you transfer binary files you must tell the server
to SET FILE TYPE BIN. You must also tell kermit on the ST to
gi (get image) or si (send image).

Q: I am using ICED.PRG to design my icons for use in my
resource file. How do I load these icons into the RCS?

A: As an example, let us consider that you are in the RCS
and have created a dialog box. Drag the icon type from the
parts box into your work area. Select the work area and
select the icon you have just moved down. Bring down the
menu under the file heading and select the open. At this
point the open item will not load a new resource but will
load in and link the data for the icon file you select in the
file selector.

Q: What documentation errors do you know about in the
Hitchhiker's Guide to the BIOS?

A: The current Hitchhiker's Guide shows the initial PC being
set from $FC0000 and the initial SP from $FC0004. These
values are reversed.

Hitchhiker's Guide is wrong in two places about the GEMDOS
call $20 to set supervisor mode. The flag to merely test the
current mode is $1 and not the $FFFFFFFF specified. The
latter will crash the system. Second, the returned value
from this function is $0 for user mode and $FF for supervisor
mode, not $0 and $1 as documented.

Q: To get started writing my program I need some more
example programs. Where can I find some?

A: We recommend that you get onto Compuserve and take a look at the Atari 16-bit SIG and also the Atari developer's SIG. For the latter, type GO PCS57, and read the membership information. We are constantly uploading example programs into the data libraries as well as other developers. These programs are available for download to your system.

Q: I am trying to redirect standard output by using the Fforce function. Gemdos is checking the handle for a number less than 6 and reporting an error. How do I redirect the console?

A: The new handle which you pass this function must be greater than 6 or less than 0. Values of -1, -2, -3 correspond to CON, AUX and PRN.

Q: How can I make a listing of my assembler program?

A: There is a way to make AS68 write a listing file to disk. To do this you must add the parameter -P and specify the output file name following the input file name with >filename. Here is an example which may work for you.

    AS68 -P -L -U file.s >list.txt

You can do this either from a batch file or you may install AS68 as a TTP and do this from the desktop.

Q: I am trying to compile the simple example C program that just uses printf to output "hello world". I keep getting undefined symbols during the compile and the object will not link. What is wrong?

A: The most likely cause of the compile errors is that you are missing the header file STDIO.H. Be sure that at the beginning of your source file that you put the following statement in:

    #include "stdio.h"

Problems with the link may be caused by not specifying all the files that are required, or placing the files in the correct order. Here is the bare minimum you should have for a C program that is not a GEM application:

    gemstart.o,(your object),gemlib,libf

Q: Why is the return key not returned by the getchar function?

A: When Digital Research wrote the 68k run time libraries for the Alcyon C compiler they used the Unix new line definition for terminating input. In a generic system the return key would generate the sequence CR LF. The run time libraries throw away the CR and count on the LF coming

through.   The only way to get around this problem is to use
the unfiltered bios or dos input from console. The scanf
function is similarly afflicted. Instead of forcing the user
to   terminate input with control-J, build a buffer of
characters terminated by a null character using the bios and
bdos.   Pass this string to sscanf which acts only on the
string and does no keyboard input.